

COL331/CSL633 Minor1 Exam
Operating Systems
Sem II, 2016-17

Answer all 6 questions

Max. Marks: 30

1. A new compiler 'hcc' is being developed. The developers of 'hcc' would like to ensure that the object files compiled with 'hcc' can be linked with the object files compiled with 'gcc'. However, the developers of 'hcc' would like to implement more optimizations, i.e., produce faster code, than 'gcc'. Answer true/false, with brief explanation:

a. The code generated by 'hcc' should be identical to the code generated by 'gcc', to be able to link with 'gcc' object files. [2]

b. 'hcc' should obey the same calling conventions as 'gcc'. If false, explain why. If true, list the 'gcc' calling conventions. [3]

2. Among all the system calls that have been discussed in class, list the system calls that are invoked (with their arguments), when you execute the following command: [5]

```
$ cat a.out | wc
```

3. Signal handling : A signal handler is specified as a function, e.g.,
void sigint_handler(int signum);

When a signal is “delivered” to a process, this function gets executed, as an “asynchronous function call”. Do all the registers need to be saved before transferring control to the signal handler function? Do only caller-save registers need to be saved? Or only callee-save registers need to be saved? Where should these registers be saved? After, the signal handler has finished executing, what should happen? [5]

4. Segmentation

- a. What is segmentation, and why is it useful? [1]
- b. How does GDT help in implementing segmentation? [2]
- c. How is GDT protected from untrusted user processes? [3]

5. Firefox uses a thread per tab, but Chrome uses a process per tab. What are the trade-offs of the two different approaches? [5]

6. What happens if the parent dies before the child exits? Is the child doomed to become a ZOMBIE forever? How can the 'init' process help; write pseudo-code for the init process to handle such orphaned child processes. [4]

